SAND2018-8766C

Outlook for Reconfigurable Accelerators

Cameron Braun

GT Center for Research in Novel Computing Hierarchies

Sandia National Labs Non Conventional Computing Technology

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

Outline

- Why accelerator-based architectures?
  - GPUs, ASICs
- Why *reconfigurable* accelerator architectures?
  - FPGA, CGRA
- Current issues in reconfigurable accelerators
  - Programmability
  - Granularity
  - Benchmarking

What is an Accelerator?

- Non CPU device used for computation
  - Not general purpose
- Examples of current popular accelerators
  - GPU (AI, graphics)
  - FPGA (DSP, packet switching)
  - ASIC (crypto, floating point)
- Can provide **order of magnitude** speedup and/or power reduction on a wide range of applications!

GPU Advantages

- Exposes massive parallelism
  - ML applications map especially well to this architecture
- Easily to obtain and integrate in current systems
- Can be reprogrammed to new applications easily

ASIC Advantages

- Can design precisely to your application and specs
- Extreme speedups possible
- Lower power than a general purpose CPU

Advantages of Accelerators

- GPUs can provide great speedups on machine learning and graphics workloads
- ASICs offer power efficient speedups on crypto and other stable workloads
- FPGAs give speedups at low power for networking and DSP

Why don’t we use accelerators for everything all the time?

Disadvantages of Accelerators

- For GPUs
  - Only good for certain types of problems
  - Power hungry

- For ASICs
  - HDLs like Verilog/VHDL are time consuming and error prone
    - Getting close to C's interface, but specialized require knowledge of architecture to write effectively
  - Refabrication every time there's a design change!

FPGA Advantages

- Design freedom/control and similar speedup potential like an ASIC

- Reconfigurability/programmability and ease of integration like a GPU
- Low power vs CPU/GPU

Industry Support

- Intel, Xilinx are finding growth in producing FPGAs
- Microsoft, Amazon provide heavy demand
  - Microsoft’s BrainWave
  - Amazon’s FPGA offerings in AWS

- However, there are issues to overcome before efficient use and widespread adoption of FPGAs happens

Issues in Reconfigurable Accelerators

1. *Programmability*
2. Granularity
3. Benchmarking

Issue 1: Programmability

Who is going to program these accelerators?

- Industry standard Verilog and VHDL aren't suitable for large scale accelerators
  - Requires domain experts to also be experts in hardware design
  - Even if they are, requires lots of programming effort
- High Level Synthesis projects have tried to bridge the gap
  - Goal: provide HDL level speedup/efficiency with (for example) normal C code

High Level Synthesis

- Benefits
  - Domain experts don't have to be experts in hardware design
  - Faster and less error prone to write

- Options currently on the market
  - **OpenCL 1.2/2.0**
  - OpenACC

OpenCL

- Interface built on top of C/C++
- Create a host program in C to distribute work to devices
- Create separate kernel program for each device to perform computations
    - Device compilation takes hours, even for very simple kernels!

OpenACC

- Commercially intended for CPU-GPU systems
- Only support FPGA via research projects at universities/nat’l labs

Issues in Reconfigurable Accelerators

1. Programmability
2. *Granularity*
3. Benchmarking

Issue 2: Granularity

- How much reconfigurability do we need?
- Traditional FPGAs allow for arbitrary bit level logic, but that isn’t what we end up using
  - Wastes space during routing and time during place
- Would be better to have an idea of the kind of units we need (ADD, shifts) and provide a connected network of those

FPGA Specialization

- Hard system for acceleration
- Host communication done in HW (link to PCIe, Xeon w/ HARP, etc.)
- ISA extensions and other small granularity functions could be enabled with lower latency/higher BW

Coarse Grain Reconfigurable Array (CGRA)

- FPGAs provide some fixed units now
  - 18x18 and 19x19 multipliers
- What makes a configuration coarse? Less arbitrary bit level logic?

Related Work: Plasticine

- Did analysis on parallel programming primitives
  - Used primitives to inform design of architecture

Plasticine Performance/Power Results

- Plasticine:
  - 11 out of 13 benchmarks had >5x performance
  - 3 had >30x
  - Power usage between 0.4x and 1.5x
  - Perf/W improvement on all apps
  - Perf improvement on all apps

Related Work: Transparent ISA Customization

- Thorough analysis of benchmark applications
- Designed a network of small reconfigurable units
  - Type A: ADD/CMP
  - Type B: MOVE/NEG
  - 7 layers deep, 7 wide at top and narrowing towards the bottom
  - Only 1 type of unit per layer

Issues in Reconfigurable Accelerators

1. Programmability
2. Granularity
3. *Benchmarking*

Issue 3: Benchmarking

- How do we compare performance with other architectures and designs?
  - MachSuite – Project to build standard set of accelerator benchmarks
    - Written to be easy to synthesize
    - Broad application areas spanned by benchmark selestions

Comparing Benchmarks

- Unlike CPUs that can run the same source code with little performance impact between them, the implementation will wildly vary the performance on different accelerator architectures
  - What do we compare?
    - Naïve models that are known to be non performant?
    - Slight modifications that don’t

Appendix

Jason Cong paper: 5 steps to get performance on FPGA using HLS

- For those unfamiliar with HLS, like most HPC programmers

Industry Shifts Towards FPGAs - Intel

- Intel End of Life on KNL host processors
  - Cited “Interest has shifted to other Intel products”
- Acquisition of FPGA maker Altera
  - Programmable Systems Group revenue up 17% from last year

Growing Investment in FPGAs - Xilinx

- Xilinx, major FPGA player
- By Market
  - 34% of revenue comes from data centers
    - 8% growth from last year
  - 48% of revenue Aerospace and defense
    - 25% growth
  - 18% of revenue Consumer and auto
    - 17% growth
- By product
  - 57% “Advanced products” (UltraScale+, UltraScale, 7 series)
    - 28% YTY growth

Real World Results - Microsoft

- Microsoft showed results from a production pilot using 1,632 servers with PCIe-based FPGA cards
  - 2x throughput, 29% lower latency, and 30% cost reduction vs. unaccelerated servers
  - ASICs could deliver ultimate efficiency
    - They simply cannot keep up with rapidly changing requirements
  - Microsoft cites the lack of an efficient optimizing compiler and related development environment as reason preventing broader FPGA use in data center applications
    - As opposed to decades of work on compilers for common CPU and GPU architectures

Amazon AWS Instances with FPGA

- Amazon has FPGA resources on AWS to allow for customizable acceleration of applications on the cloud

What about CUDA?

- Allows easy parallelism across Nvidia GPU resources
- But ONLY Nvidia GPUs

FPGA vs. GPU for DNN

- GPUs traditionally compute in 32 bit chunks
- FPGAs can take advantage of lower precision data types that are the focus of new research
  - Good for inference, at least